



## Definición de mapas

### Transcripción

[00:00] Hola, hola. Vamos a iniciar nuestra clase 18. Para esto vamos a duplicar nuestra clase 17, vamos a colocar clase 18, y manos a la obra. Vamos a hacer lo siguiente. Vamos a remover toda la parte de iterator y vamos a ponernos la siguiente situación. En teoría el código de alumno es único, entonces yo puedo hacer búsquedas por código de alumno.

[00:28] Imaginémonos que yo quiero el alumno con código 003. Utilizando por ejemplo string sería una cosa parecida a esta de aquí:

`getAlumnos().stream().filter`, y aquí colocamos alumno, vamos a colocar aquí letra A de alumno. (`a -> a.getCodigo().equalsIgnoreCase` y aquí dijimos que queremos "003").

[01:16] Como buena práctica, como ya sabemos el valor, y puede existir imaginémonos caso hipotético que código sea nulo, cuando seleccione qué código sería nulo `.get equalsIgnoreCase()`, va a dar `nullPointer`, entonces hacemos aquí y así evitamos el `nullPointer` porque aquí va a ser nulo. Una vez hecho esto utilizamos aquí `findFirst`. ¿Qué quiere decir? Quiero el primero, en teoría sería el único.

[01:51] Esto de aquí me devuelve un optional, este optional sería un optional ¿de qué? De alumno. Y aquí sería alumno. Esto después lo van a ver en las clases de Java 8 por ejemplo. Aquí en el optional, de Java 8 para el frente en realidad, aquí en el optional podemos hacer lo siguiente. Como es un opcional quiere decir: puede tener como no puede tener, como su nombre lo dice.

[02:26] Ahí utilizamos el método `isPresent`. En este método `isPresent`, podemos hacer lo siguiente. Ahí si es `true`, quiere decir que existe, entonces `System.out.println(alumno.get)` quiero que me traigas a alumno. Y ejecutamos. ¿Qué sucedió? Me trajo Juan Carlos, pero aquí no se pudo ver por qué dejé aquí `forEach`. Vamos a ejecutar de nuevo. Y Juan Carlos. Perfecto.

[03:05] Supongamos que yo no hubiese usado el `isPresent`, y vamos a colocar aquí "013". ¿Qué cosa va a suceder? Nulo. ¿Por qué? Porque no existe elemento. Por eso siempre es bueno utilizar esta parte de aquí de `isPresent`. Listo. Aquí tenemos una forma, pero como lo dije una vez, esto de aquí es único. Entonces si es único, ¿sería bueno utilizarlo como una lista, como un `collection`? Tal vez no.

[03:53] Para esto existen los mapas. ¿Qué son los mapas? Tenemos el mapa mundial, ¿no? En el mapa mundial sabemos dónde está Brasil, donde está Chile, Argentina, Perú y en qué lado está ubicado en el mapa. Entonces, lo mismo tenemos en la situación en Java. Tenemos la opción de poder utilizar mapas.

[04:18] Para esto, vamos a hacer lo siguiente. Aquí, en lugar, ya dejamos aquí la colección de alumnos y vamos a ver aquí un mapa. Por ejemplo colocamos `Map`. El mapa viene de `Java.util` también pero no hace parte de `collection`, es una parte bien importante. Aquí tenemos todo mapa tiene su `key` y tiene su valor. Entonces ¿el `key` qué sería? Lo que es único.

[04:56] En nuestro caso el `key` es el código, que es un `string`. Y el valor puede ser el alumno. Una vez hecho esto, tenemos aquí `alumnoMap = new HashMap`, que es el `Map` más tradicional que utilizamos. Una vez hecho esto, imaginémonos aquí en nuestro método `add(alumno)`, vamos a utilizar el `this`. Vamos a ver el alumno cómo lo puse aquí, `alumnoMap.put` que sería nuestro método para adicionar, el `put`.

[05:52] Aquí colocamos el key que sería el `alumno.getCodigo()`, y `alumno`. Listo. Entonces nuestro `add` va a adicionar de dos formas los alumnos. Uno, utilizando el `hashSet` y otro utilizando el `hashMap`. Aquí está `addAlumno`. Ahora, utilizando esta parte así, ya tenemos el método `getAlumnos`, entonces vamos a colocar aquí el método. Vamos a implementar aquí nuestro getter de `alumnoMap`.

[06:38] Entonces viendo de esa forma, ¿que haríamos? Haríamos así: `curso1.getAlumnoMap().get` que sería por el key. ¿Cuál es nuestro key? "003". Esto me devuelve ¿qué cosa? El value. ¿Cuál es el value? El `alumnoMap`. Y aquí utilizamos `System.out.println` y aquí utilizamos `alumnoMap` y listo. Ejecutamos.

[07:34] Imprimió dos veces Juan Carlos, de aquí y de aquí. ¿Cuál fue más rápido? Utilizando el mapa. Y después vamos a ver por qué también es bueno utilizar el mapa, pero esto sería para nuestra siguiente clase, esto sería todo por ahora. Muchas gracias.